Consider the Figure given in the next pages, in which the ID/EX stages have been simplified, e.g., the sign-extended immediate and branch logic are not drawn.

Also consider the following sequence of 8 instructions for the pipelined MIPS. Assume the pipeline is initially empty.

```
start:   lw  $s1,0($a0)
        lw  $s2,4($a0)
        add $t1,$s1,$s2
        sub $t2,$s1,$s2
        add $t3,$t1,$t2
        sub $t4,$t1,$t2
        add $t5,$t3,$t4
        sw  $t5,8($a0)
```
Although the ID and EX stages have been simplified—the sign-extended immediate and branch logic are missing—this drawing gives the essence of the forwarding hardware requirements.

Elaboration:
Regarding the remark earlier about setting control lines to 0 to avoid writing registers or memory: only the signals RegWrite and MemWrite need be 0, while the other control signals can be don't cares.

4.8 Control Hazards
Thus far, we have limited our concern to hazards involving arithmetic operations and data transfers. However, as we saw in Section 4.5, there are also pipeline hazards involving branches. Figure 4.61 shows a sequence of instructions and indicates when the branch would occur in this pipeline.

There are a thousand hacking at the branches of evil to one who is striking at the root.
Henry David Thoreau, Walden, 1854

4.8 Control Hazards